

Projeto de Bases de Dados

Parte 2

Grupo: 50			
Número	Nome	Contribuição	Esforço (em Horas)
93695	Catarina Sofia dos Santos Sousa	34%	3h
93743	Nelson Alexandre Geada Trindade	33%	3h
93754	Rodrigo Rodrigues Major	33%	3h

Sala: 1-29

Turno: 4ªFeira – 15h00

Docente: Carlota De Oliveira Lopes Dias

Modelo Relacional

Observação (ID, #Doente, Valor)

Intervenção (ID, data observação)

- ID: FK(Observação)

Acto_Médico (ID, #número)

- ID: FK(Intervenção)

Medição (ID, tipo)

- ID: FK(Observação)

Análise_Lab (ID)

- ID: FK(Medição)

Leitura_Local (ID)

- ID: FK(Medição)

Médico (#Cédula, Nome, Especialidade)

Agenda (Data, Hora)

Instituição (Nome, Morada)

Protocolo (NrProtocolo, data_homologação, descrição)

Consulta (#Cédula, Data, Hora, ID, #numero, especialidade)

- #Cédula: FK(Médico)
- Data, Hora: FK (Agenda.Data, Agenda.Hora)
- ID, #numero: FK (Acto_Médico.ID, Acto_Médico.#numero) NOT NULL

TemplateAnálise (NrProtocolo, Nome, Morada, ID)

- NrProtocolo: FK(Protocolo)
- Nome, Morada: FK(Instituição.Nome, Instituição.Morada)
- ID: FK(Análise_Lab)

Validado (Nome, Morada, NrProtocolo, IDTemplateAnálise, #numero, IDActo_Médico, data)

- Nome, Morada, NrProtocolo, IDTemplateAnálise: FK(TemplateAnálise.Nome, TemplateAnálise.Morada, TemplateAnálise.NrProtocolo, TemplateAnálise.ID)
- #numero, IDActo_Médico: FK(Acto_Médico.#numero, Acto_Médico.ID) NOT NULL

Faz (#Cédula, NrProtocolo, Nome, Morada, ID)

- #Cédula: FK(Médico)
- NrProtocolo, Nome, Morada, ID: FK (TemplateAnálise.NrProtocolo, TemplateAnálise.Nome, TemplateAnálise.Morada, TemplateAnálise.ID)

Segundo (NrProtocolo, ID)

- NrProtocolo: FK(Protocolo)
- ID: FK(Leitura_Local)

Restrições de Integridade

-> ID nunca existe em Intervenção e Medição ao mesmo tempo;

-> ID tem de existir em Intervenção ou em Medição;

-> para todo o ID da tabela Análise_Lab tem de existir pelo menos uma entrada na tabela TemplateAnálise;

(RI-1): Para cada Médico cuja #Cédula está presente na tabela Consulta, a sua especialidade tem de ser igual à especialidade presente também na tabela Consulta;

(RI-2): Para cada quarteto NºProtocolo, Nome, Morada e ID presente na tabela TemplateAnalise tem de existir uma entrada na tabela Faz ou na tabela Validado;

(RI-3): Para todo o #numero na tabela Acto Médico, não pode existir uma entrada nas tabelas TemplateAnálise e Consulta que seja igual;

Álgebra Relacional

Pergunta 1

$$\pi_{\#Cédula}(\sigma_{Data = "20-11-2020" \wedge Hora = "14:00"}(Consulta))$$

Pergunta 2

$$\begin{aligned} Aux &\leftarrow \pi_{\#Doente, ID}(Observação \bowtie AnaliseLab) \\ Counter &\leftarrow \left(\rho_{Counter(2 \rightarrow Analises)}(\#Doente G_{count(ID)}(Aux)) \right) \\ \pi_{\#Doente} &\left(\sigma_{Analises = MAX} \left(Counter X \rho_{R(1 \rightarrow MAX)}(G_{max(Analises)}(Counter)) \right) \right) \end{aligned}$$

Pergunta 3

$$\begin{aligned} AuxProtocolos &\leftarrow \left(\rho_{AuxProtocolos(1 \rightarrow Limite)}(G_{count(Nr.Protocolo)}(Protocolo)) \right) \\ AuxDoentes &\leftarrow \left(\pi_{(\#Doente, ID, NrProtocolo)}(Observação \bowtie TemplateAnalise) \right) \\ ProtocolosAnalise &\leftarrow \left(\rho_{ProtocolosAnalise(3 \rightarrow counter)}(\#Doente, ID G_{count-distinct(NrProtocolo)}(AuxDoentes)) \right) \\ \pi_{\#Doente} &\left(\sigma_{AuxProtocolos.Limite = ProtocolosAnalise.counter}(AuxProtocolos X ProtocolosAnalise) \right) \end{aligned}$$

Pergunta 4

$$\begin{aligned} Aux &\leftarrow \left(\rho_{Aux(3 \rightarrow atosMedicos)}(Data, \#Cédula G_{count(\#numero)}(Consulta)) \right) \\ \pi_{\#Cédula} &\left(Data, \#Cédula G_{max(atosMedicos)}(Aux) \right) \end{aligned}$$

Pergunta 5

$$\left(\pi_{\#Cédula, Data}(Consulta) \right) \div \left(\pi_{Data}(\sigma_{Data \geq "1-1-2020" \wedge Data \leq "2-2-2020"}(Consulta)) \right)$$

SQL

Pergunta 1

```
SELECT m.cedula  
FROM Médico m, Consulta c  
WHERE m.cedula = c.cedula  
      AND c.data = "20-11-2020"  
      And c.hora = "14:00";
```

Pergunta 2

```
With tempr as (  
  SELECT o.Doente as Nome_Doente, COUNT(DISTINCT a.ID) as num_Analises  
  FROM Observação o, AnáliseLab a  
  WHERE a.ID = o.ID  
  GROUP BY Nome_Doente  
  ORDER BY Nome_Doente ASC, num_Analises DESC  
)  
SELECT Nome_Doente  
From tempr sub  
WHERE sub.num_Analises = (SELECT MAX(sub.num_Analises) FROM tempr sub);
```

Pergunta 3

```
With tempr as (  
  SELECT o.Doente as Doente, COUNT(DISTINCT t.NrProtocolo) as CounterProtocolos  
  FROM TemplateAnalyse t NATURAL JOIN Observação o  
  GROUP BY Doente  
  ORDER BY CounterProtocolos DESC, Doente ASC  
)  
SELECT Doente  
From tempr sub  
WHERE sub.CounterProtocolos < (SELECT COUNT(p.NrProtocolo) FROM Protocolo p);
```